In the Claims

Please replace all prior versions, and listings, of claims in the application with the following list of claims:

1. (Currently amended) A computer system for executing instructions having assigned guard indicators, which system comprises instruction supply circuitry, at least one pipelined execution unit for receiving instructions from the supply circuitry together with at least one guard indicator selected from a set of guard indicators, said execution unit including a master guard value store containing master values for the guard indicators in said set of guard indicators, and circuitry for resolving the guard value of the or each guard indicator in the execution pipeline and providing a signal to indicate whether the pipeline is committed to the execution of the instruction, said system including an emulator having watch circuitry for effecting a watch on selected instructions supplied to the execution pipeline and ~~synchronising~~ synchronizing circuitry for correlating resolution of the guard indicator of each selected instruction with a program count for that instruction.

2. (Original) A computer system according to claim 1 in which said watch circuitry is arranged for watching instruction fetch addresses supplied to a program memory and providing an indication that the instruction is selected prior to decoding the instruction.

3. (Currently amended) A computer system according to claim 1 in which the emulator includes data watch circuitry for providing an output signal to said ~~synchronising~~ synchronizing circuitry to indicate detection of a selected data value output by the execution pipeline.

4. (Original) A computer system according to ~~preceding~~ claim 1 in which the emulator includes load/store detection circuitry to indicate execution of a load or store operation by said execution pipeline.

5. (Original) A computer system according to claim 1 in which the emulator includes memory access address watch circuitry to indicate detection of a selected memory access address by said execution pipeline.

6. (Currently amended) A computer system according to claim 1 in which said ~~synchronising~~ synchronizing circuitry comprises a plurality of multivalue buffers each arranged to hold successive values in an ordered sequence.

7. (Original) A computer system according to claim 6 in which said buffers comprise a plurality of FIFOs.

8. (Original) A computer system according to claim 7 in which said FIFOs include at least one for the program count of selected instructions.

9. (Original) A computer system according to claim 7 in which said FIFOs include at least one for commit indicators after resolution of the guard values of instructions supplied to the execution pipeline.

10. (Original) A computer system according to claim 7 in which said FIFOs include at least one for indicating whether load/store instructions have been executed in the execution unit.

11. (Original) A computer system according to claim 7 in which said FIFOs include at least one for indicating execution of a data memory access instruction where the data and/or address provide the hit with a watched value of the data and/or address.

12. (Original) A computer system according to claim 1 in which a plurality of execution units are provided in parallel and a plurality of instructions are dispatched simultaneously to respective execution pipelines.

13. (Original) A computer system according to claim 12 in which all guard indicators of instructions dispatched simultaneously to a plurality of pipelines are supplied to one or more pipelines having access to said master guard value store when the watching circuitry of said emulator has selected said instructions.

14. (Currently amended) A method of executing instructions in a computer system, said instructions having assigned guard indicators, and effecting a debugging watch on execution of said instructions, which method comprises supplying a plurality of instructions to at least one pipelined execution unit, resolving the guard value of each instruction in the execution unit to provide a commit signal if the instruction is executed, and effecting a debugging watch by selecting instructions in an instruction fetch operation and providing a program count signal for said selected instructions, storing the program count in a sequential buffer, and storing in a further sequenced buffer commit signals derived from resolution of the guard value in the execution pipeline thereby ~~synchronising~~ synchronizing the program count with resolution of the guard value for the selected instruction.

15. (Original) A method according to claim 14 in which a plurality of other parameters are watched by the debugging routine and detection of the parameters is stored in further buffers which hold entries in an ordered sequence for synchronisation with the commit signals and program counts.

16. (Original) A method according to claim 14 in which the occurrence of events which are watched by the debugging routine are held in respective first in first out buffers.

17. (Original) A method according to claim 12 in which the debugging routine provides a trace or profile of events related to the program count of a watched instruction for which a commit signal was provided by the execution pipeline.

18. (Original) A method according to claim 12 in which a plurality of instructions are supplied simultaneously to a plurality of parallel execution pipelines the resolution of guard values for each of the instructions being effected during passage through the execution pipeline.